Flot maximum avec contrainte de délai proportionnel

Pierre Bonami, Dorian Mazauric, Yann Vaxès

Flot maximum avec contrainte de délai proportionnel

Pierre Bonami[1,2] and Dorian Mazauric[1] and Yann Vaxès[1]

[1] *Aix-Marseille Université, CNRS, LIF UMR 7279, 13000, Marseille, France*
[2] *IBM ILOG CPLEX, Madrid, Espagne*

Étant donnés un réseau et un ensemble de paires source-destination (connexions), nous considérons le problème de maximiser la somme des flots avec des contraintes de délai proportionnel. Dans cet article, le délai pour traverser un lien est proportionnel au flot total supporté par ce lien. Si une connexion supporte un flot non nul, alors la somme des délais tout au long du chemin correspondant à cette connexion doit être plus petite qu'une certaine borne donnée. Ces contraintes de délai sont des contraintes de type "on-off" car si une connexion supporte un flot nul, alors il n'y a pas de contrainte pour cette connexion. La difficulté de ce problème réside dans le choix des connexions supportant un flot non nul.

Nous prouvons une approximation générale en utilisant la programmation linéaire. Nous montrons ensuite un schéma d'approximation polynomial lorsque le graphe d'intersection des chemins a une largeur arborescente bornée. Nous prouvons enfin que le problème est NP-difficile même lorsque le réseau est un arbre.

Keywords: flot maximum, contraintes de délai "on-off", PTAS, programmation dynamique, largeur arborescente bornée, programmation linéaire, NP-difficile.

1 Introduction

Contexte et motivations. Le problème de flot à plusieurs commodités a été largement étudié dans la littérature. Étant donnés un réseau, un ensemble de capacités sur les arêtes et un ensemble de demandes (commodités), le problème consiste à déterminer un flot satisfaisant toutes les demandes et respectant les contraintes de capacité et de conservation de flots. La version entière est un problème NP-complet [EIS76] même pour deux commodités et des capacités unitaires (rendant le problème fortement NP-complet dans ce cas). Cependant, si un flot fractionnaire est autorisé, alors le problème devient polynomial car il peut se formuler comme un programme linéaire [AMO93].

Les opérateurs des réseaux de télécommunication doivent satisfaire certaines exigences concernant la qualité de service pour leurs clients. Un des paramètres les plus importants est le délai de bout en bout d'une unité de flot entre un nœud source et un nœud destination. Cette exigence n'est pas prise en compte dans le problème de flot à plusieurs commodités. Le délai sur un lien dépend du volume de flot supporté par ce lien ; classiquement cela est modélisé par une fonction convexe. Le délai de bout en bout pour une demande et un chemin associé est la somme des délais de tous les liens du chemin. Certains articles se sont focalisés sur la minimisation de la moyenne du délai de bout en bout. Ce problème consiste à minimiser une fonction convexe avec des contraintes linéaires [OMV97] et peut être résolu en utilisant la programmation semi définie [TAG03]. D'autres articles se sont focalisés sur la recherche d'un flot à plusieurs commodités satisfaisant les demandes et minimisant le plus grand délai de bout en bout [CSM07]. Comme les auteurs de [BAO06], nous pensons qu'un problème plus réaliste consiste à ajouter des contraintes strictes sur les délais de bout en bout pour toutes les connexions. En effet, il y a différentes classes de service dans les réseaux de communication et, pour chacune d'entre elles, il est crucial de respecter un certain niveau de qualité de service, c'est-à-dire respectant un seuil pour le délai de bout en bout.

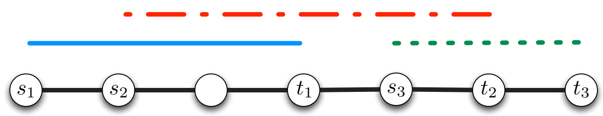

FIGURE 1: Le graphe $G = (V, E)$ est un chemin composé de sept nœuds et $\alpha_e = 1$ pour tout $e \in E$. Il y a trois paires source-destination : (s_1, t_1), (s_2, t_2) et (s_3, t_3). Les trois chemins sont représentés au dessus de G.

Dans cet article, nous nous focalisons sur un problème de flot à plusieurs commodités dans lequel chaque arête a une fonction de latence proportionnelle qui décrit le délai commun pour les flots utilisant cette arête comme une fonction du flot total sur cette arête. Nous nous intéressons au problème de maximiser la somme du flot sous des contraintes de délai proportionel. Nous autorisons des flots fractionnaires pour toutes les connexions. Comme mentionné précédemment, nous avons une contrainte sur le délai de bout en bout pour chaque demande. Mais si un chemin associé à une connexion supporte un flot nul, alors la contrainte n'est pas considérée. Ces contraintes de type "on-off" rendent le problème beaucoup plus difficile que la résolution d'un programme linéaire [HBCO12].

Modélisation du problème et définitions. Soit $G = (V, E)$ un graphe connexe (représentant un réseau) avec un coefficient α_e pour toute arête $e \in E$. Soient $\{(s_1, t_1), \ldots, (s_m, t_m)\}$ un ensemble de m paires source-destination (connexions). Soit $\mathcal{P} = \{P_1, \ldots, P_m\}$ un ensemble de m chemins dans G. Le chemin P_i correspond à la paire (s_i, t_i) pour tout $i = 1, \ldots, m$. Sans perte de généralité, nous supposons qu'il y a un unique chemin pour chaque paire source-destination. Nous notons x_i le flot supporté par le chemin P_i pour tout $i = 1, \ldots, m$. Nous supposons que le délai τ_e pour traverser l'arête $e \in E$ est proportionnel au flot total $\sum_{i:e \in E(P_i)} x_i$ supporté par l'arête e, i.e. $\tau_e = \alpha_e \sum_{i:e \in E(P_i)} x_i$. Soit $\lambda > 0$. Pour tout $i = 1, \ldots, m$, nous exigeons pour le chemin P_i que, si $x_i > 0$, alors le délai de bout en bout $\sum_{e \in E(P_i)} \tau_e$ est au plus λ. Par une mise à l'échelle des coefficients α_e (division par λ), nous pouvons supposer que $\lambda = 1$. Un flot à plusieurs commodités $x = (x_1, \ldots, x_m)$ satisfait la contrainte de latence si pour tout $j = 1, \ldots, m$ tel que $x_j > 0$, la contrainte suivante pour le chemin P_j est respectée : $\sum_{e \in E(P_j)} \tau_e \leq 1$. En notant $\beta_{i,j} := \sum_{e \in E(P_i) \cap E(P_j)} \alpha_e$, cette contrainte revient à : $\sum_{i=1}^{m} \beta_{i,j} x_i \leq 1$. Le problème de flot à plusieurs commodités avec contrainte de délai (problème FCD) consiste à trouver parmi les solutions satisfaisant ces contraintes, une solution de valeur $\sum_{i=1}^{m} x_i$ maximale. En d'autres termes, le problème revient à résoudre :

$$\left\{ \begin{array}{llll} \text{Max} & \sum_{i=1}^{m} x_i & & \\ & \sum_{i=1}^{m} \beta_{i,j} x_i \leq 1 & j = 1, \ldots, m & x_j > 0 \\ & x_i \geq 0 & i = 1, \ldots, m & \end{array} \right.$$

La difficulté du problème FCD réside dans le choix des chemins supportant un flot non nul. En effet, si l'ensemble des chemins $\mathcal{P}^* \subseteq \mathcal{P}$ supportant un flot non nul est donné, alors le problème devient polynomial car il se ramène à résoudre un programme linéaire.

Le graphe d'intersection des chemins H a pour ensemble de sommets $V(H) = \{h_1, \ldots, h_m\}$ correspondant à l'ensemble des chemins $\mathcal{P} = \{P_1, \ldots, P_m\}$. Pour $i, j \in \{1, \ldots, m\}$, $i \neq j$, il y a une arête $\{h_i, h_j\} \in E(H)$ entre deux sommets $h_i \in V(H)$ et $h_j \in V(H)$ si, et seulement si, il existe une arête $e \in E$ telle que $e \in E(P_i) \cap E(P_j)$, c'est-à-dire lorsque P_i et P_j partagent au moins une arête.

Exemple. Soit le chemin $G = (V, E)$ de la Figure 1 et les trois paires source-destination (s_1, t_1), (s_2, t_2) et (s_3, t_3). Le chemin P_i est l'unique chemin simple entre s_i et t_i dans G pour tout $i \in \{1, 2, 3\}$. Notons que H est un chemin composé de trois sommets. Posons $\alpha_e = 1$ $\forall e \in E$. Le problème revient à maximiser $x_1 + x_2 + x_3$ sous les contraintes $x_1(3x_1 + 2x_2) \leq x_1$, $x_2(2x_1 + 4x_2 + x_3) \leq x_2$, $x_3(x_2 + 2x_3) \leq x_3$ et $x_1, x_2, x_3 \geq 0$. La solution $x^* = (1/3, 0, 1/2)$ est optimale. La connexion (s_2, t_2) supporte un flot nul, c'est-à-dire $x_2 = 0$, expliquant pourquoi $2x_1 + 4x_2 + x_3 = 7/6 > 1$.

Nos contributions. Nous analysons tout d'abord la qualité de la solution obtenue par le programme linéaire pour une variante du problème FCD pour laquelle toutes les contraintes de délai doivent être satisfaites même pour les connexions qui ne supportent pas de flot non nul. Nous en déduisons un algorithme d'approximation polynomial. En particulier, cela donne une L-

approximation avec L la taille d'un plus long chemin de $\mathcal{P}$ et une 2-approximation lorsque G est un chemin. Dans le cas des chemins, nous donnons une autre 2-approximation qui s'exécute en temps linéaire. Nous prouvons ensuite un schéma d'approximation polynomial lorsque le graphe d'intersection des chemins H a une largeur arborescente bornée. Pour cela, nous développons un algorithme exact de programmation dynamique pour une variante du problème pour lequel le volume de flot possible pour toute connexion est pris dans un ensemble fini donné X. Enfin, nous prouvons que le problème est NP-difficile même lorsque G est un arbre. L'intégralité de nos résultats et de nos preuves se trouve dans [BMV14].

2 Algorithme d'approximation général

Pour tout $k = 1, \ldots, m$, soit $S_k^* \subseteq \{1, \ldots, m\} \setminus \{k\}$ un ensemble d'indices de cardinalité maximale tel que pour tout $i \in S_k^*$, il existe $e \in E(P_k) \cap E(P_i)$ telle que pour tout $j \in S_k^* \setminus \{i\}$, $e \notin E(P_j)$. Soit $|S^*| = \max_{i=1,\ldots,m} |S_i^*|$. Soit x^* une solution optimale pour le problème FCD et soit x une solution optimale pour le problème où toutes les contraintes de délai sont satisfaites (même pour les connexions ne supportant pas de flot non nul). Nous avons prouvé que $\sum_{i=1}^{m} x_i \geq \sum_{i=1}^{m} x_i^* / |S^*|$. Comme x peut être obtenu en résolvant un programme linéaire, nous en déduisons dans le Théorème 1 un algorithme d'approximation.

Théorème 1 *Il existe un algorithme polynomial avec facteur d'approximation $|S^*|$ pour le problème FCD.*

En particulier, cela donne un algorithme polynomial avec un facteur d'approximation $L = \max_{i=1,\ldots,m} |E(P_i)|$ et un algorithme polynomial avec un facteur d'approximation 2 lorsque le graphe G est un chemin. Pour le cas du chemin, nous avons prouvé une autre 2-approximation basée sur le calcul d'une meilleure solution indépendante (deux chemins qui supportent un flot non nul n'ont pas d'arête commune). Cela revient à calculer un ensemble indépendant de poids maximum de H. Si le réseau est un chemin, alors H est un graphe d'intervalles et une solution indépendante optimale peut être calculée en temps linéaire.

3 PTAS lorsque H a une largeur arborescente bornée

Nous définissons tout d'abord une variante du problème : le problème de flot discret à plusieurs commodités avec contrainte de délai (problème FDCD). Le seul changement est que, étant donné un ensemble fini X de valeurs positives contenant zéro, le flot $x_i \in X$ pour tout $i = 1, \ldots, m$. La solution $x = (1/3, 0, 1/3)$ est une solution optimale pour cette variante pour l'instance de la Figure 1 avec $X = \{0, 1/3, 2/3, 1\}$. Nous prouvons dans le Lemme 1 un algorithme exact de programmation dynamique pour le problème FDCD.

Lemme 1 *Soit X un ensemble fini de valeurs positives contenant zéro. Il existe un algorithme exact de complexité $O(m|X|^{tw(H)+1})$ pour le problème FDCD avec $tw(H)$ la largeur arborescente de H.*

Soient $x_{max} = \max_{i=1,\ldots,m} 1/\sum_{e \in E(P_i)} \alpha_e$ et $x_{min} = \min_{i=1,\ldots,m} 1/\sum_{e \in E(P_i)} \alpha_e$. Nous prouvons dans le Théorème 2 un schéma d'approximation polynomial pour le problème FCD lorsque H a une largeur arborescente bornée et lorsque x_{max}/x_{min} est borné.

Théorème 2 *Soient $b, t \geq 1$ des constantes. Pour tout $\varepsilon > 0$, il existe un algorithme polynomial avec facteur d'approximation $1 + \varepsilon$ pour le problème FCD lorsque $tw(H) \leq t$ et $x_{max}/x_{min} \leq b$.*

Nous déduisons du Théorème 2 qu'il existe un schéma d'approximation polynomial lorsque G est un arbre et lorsque x_{max}/x_{min}, Δ_G et χ_G sont bornés par des constantes, avec Δ_G le degré maximum de G et χ_G le nombre maximum de chemins qui partagent une même arête. Cela montre aussi l'existence d'un schéma d'approximation polynomial lorsque G est un chemin et lorsque x_{max}/x_{min} et χ_G sont bornés par des constantes. Cependant, la complexité du problème reste ouverte lorsque G est un chemin si χ_G n'est pas borné par une constante. Plus précisément, nous ne savons pas si le problème est NP-difficile et s'il existe un algorithme polynomial avec un rapport d'approximation strictement plus petit que 2. Enfin, les résultats précédents sont valides même si les seuils de délai de bout en bout sont quelconques (si $x_i > 0$, alors $\sum_{e \in E(P_i)} \tau_e \leq \lambda_i \ \forall i = 1, \ldots, m$).

4 Le problème est NP-difficile même lorsque G est un arbre

Le problème de multiflot entier maximum est NP-difficile dans les arbres [GVY97]. Nous prouvons dans le Théorème 3 que le problème FCD est également NP-difficile dans les arbres. Contrairement à tous nos autres résultats, nous utilisons ici des capacités sur les arêtes de l'arbre. Le problème FCD a été prouvé NP-difficile dans [BAO06] pour une fonction de délai convexe mais la question a été laissée ouverte lorsque le graphe G est un arbre. En effet, leur preuve implique un choix entre deux chemins différents pour chaque connexion et force donc le graphe G à contenir des cycles. Notons que dans la preuve du Théorème 3, nous avons $tw(H) = \Omega(V(H))$. Nous ne savons donc pas s'il existe un algorithme polynomial avec facteur d'approximation constant pour la classe d'instances considérée.

Théorème 3 *Le problème FCD est NP-difficile même si le graphe G est un arbre.*

5 Extensions et travaux futurs

Pour conclure, nous mentionnons quelques pistes de recherche et problèmes ouverts. Quelle est la complexité du problème FCD quand G est un chemin ? Un premier problème ouvert intéressant est de déterminer la complexité du problème lorsque G est un chemin tel que toutes les connexions partagent une arête $e \in E$ (H est un graphe complet). Est-ce qu'il existe un algorithme polynomial garantissant un facteur d'approximation strictement plus petit que 2 quand le graphe est un chemin avec χ_G non borné ? Est-ce qu'il existe un algorithme exact polynomial lorsque $tw(H)$ est bornée ? Est-ce qu'il existe des classes d'instances avec $tw(H)$ non bornée qui admettent un schéma d'approximation polynomial ? Est-ce que le problème FCD est dans APX ?

Un axe intéressant est le développement de règles de réduction polynomiales à partir, par exemple, de solutions obtenues pour le problème où toutes les contraintes de délai sont satisfaites (même pour une connexion avec un flot nul). De telles solutions sont obtenues en temps polynomial par la résolution d'un programme linéaire. Considérons un chemin $P_i \in \mathcal{P}$. Notons $LP(\mathcal{P})$ la valeur optimale et notons $LP(\mathcal{P} \setminus \{P_i\})$ la valeur optimale de l'instance sans le chemin P_i. Pour quelles classes d'instances pouvons-nous supprimer P_i si $LP(\mathcal{P} \setminus \{P_i\}) > LP(\mathcal{P})$? Pour quelles classes d'instances $LP(\mathcal{P} \setminus \{P_i\}) < LP(\mathcal{P})$ implique l'existence d'une solution optimale telle que $x_i > 0$?

Références

[AMO93] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network flows : theory, algorithms, and applications*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[BAO06] Walid Ben-Ameur and Adam Ouorou. Mathematical models of the delay constrained routing problem. *Algorithmic Operations Research*, 1(2), 2006.

[BMV14] Pierre Bonami, Dorian Mazauric, and Yann Vaxès. Maximum flow under proportional delay constraint. Technical report, Aix-Marseille Université, CNRS, LIF UMR 7279, January 2014. available at : http ://pageperso.lif.univ-mrs.fr/∼dorian.mazauric/BMV14b.pdf.

[CSM07] José R. Correa, Andreas S. Schulz, and Nicolás E. Stier Moses. Fast, fair, and efficient flows in networks. *Operations Research*, 55(2) :215–225, 2007.

[EIS76] Shimon Even, Alon Itai, and Adi Shamir. On the complexity of timetable and multicommodity flow problems. *SIAM J. Comput.*, 5(4) :691–703, 1976.

[GVY97] N. Garg, V.V. Vazirani, and M. Yannakakis. Primal-dual approximation algorithms for integral flow and multicut in trees. *Algorithmica*, 18(1) :3–20, 1997.

[HBCO12] Hassan Hijazi, Pierre Bonami, Gérard Cornuéjols, and Adam Ouorou. Mixed-integer nonlinear programs featuring "on/off" constraints. *Comp. Opt. and Appl.*, 52(2) :537–558, 2012.

[OMV97] A. Ouorou, P. Mahey, and J.-Ph. Vial. A survey of algorithms for convex multicommodity flow problems, 1997.

[TAG03] C. Touati, E. Altman, and J. Galtier. Semi-definite programming approach for bandwidth allocation and routing in networks. *Game Theory and Applications*, 9 :169–179, 2003.